

FLEXIBLE DATA RATE MATCHING APPARATUS AND METHOD
IN A DATA COMMUNICATION SYSTEM

PRIORITY

5

This application claims priority to an application entitled "Flexible Data Transmission Apparatus and Method in a Data Communication System" filed in the Korean Industrial Property Office on April 21, 2000 and assigned Serial No. 2000-22039; an application entitled "Flexible Data Transmission Apparatus and Method in a Data Communication System" filed in the Korean Industrial Property Office on April 24, 2000 and assigned Serial No. 2000-21672; an application entitled "Flexible Data Transmission Apparatus and Method in a Data Communication System" filed in the Korean Industrial Property Office on April 26, 2000 and assigned Serial No. 2000-22295; and an application entitled "Flexible Data Transmission Apparatus and Method in a Data Communication System" filed in the Korean Industrial Property Office on April 27, 2000 and assigned Serial No. 2000-22521, the contents of all of which are hereby incorporated by reference.

20

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates generally to a data communication system, and in particular, to an apparatus and method for matching a frame having coded symbols flexibly determined according to variation of a data rate to an interleaver size.

2. Description of the Related Art

In a radio communication system such as a satellite system, an ISDN (Integrated Services Digital Network) system, a digital cellular system, a W-

CDMA (Wideband Code Division Multiple Access) system, a UMTS (Universal Mobile Telecommunications System) system, and IMT-2000 (International Mobile Telecommunication-2000) system, a channel coding scheme chiefly uses a convolutional code, and a linear block code for which a single decoder is used.

- 5 Symbols coded by such a channel coding scheme are generally interleaved by a channel interleaver.

A typical channel interleaver was designed to perform interleaving by receiving a frame having coded symbols, the number of which is identical to an 10 interleaver size per frame. However, a recent FDRT (Flexible Data Rate Transmission) channel interleaver performs interleaving by receiving a frame having coded symbols, the number of which is different from an interleaver size per frame.

- 15 FIG. 1 illustrates a non-FDRT (or fixed data rate transmission) channel interleaver which performs interleaving by receiving a frame having coded symbols, the number of which is identical to an interleaver size.

Referring to FIG. 1, in the non-FDRT mode where the data rate of a 20 channel is fixed, the number L of coded symbols per frame, input to a channel interleaver 100, is always equal to an interleaver size N. For example, RC (Radio Configuration) used in the IMT-2000 communication system includes various transmission channel types such as RC1, RC2, RC3, RC4, RC5, RC6, RC7, RC8 and RC9, having different data frame size, code rate and interleaving mode.

- 25 Accordingly, in the non-FDRT mode, only a predetermined fixed data rate is used.

FIG. 2 illustrates an example of a coded symbol frame format transmitted in the non-FDRT mode.

Referring to FIG. 2, if it is assumed that a physical channel is set to an RC3 data rate=19.2Kbps, then the size of the channel interleaver 100 shown in FIG. 1 becomes N=1536. Data transmitted as 19.2Kbps in 20msec periods have 384bit per sec, and data after channel encoder which has 1/4 as R are to have 5 1536bit per sec. At this moment, if the user desires to transmit the frame at a data rate 20Kbps, a base station and a mobile station determine a data rate 38.4Kbps out of available data rates higher than the desired data rate 20Kbps, in an initial negotiation process. This is because the data rate 38.4Kbps is the least data rate higher than 20Kbps. When the data rate is set to 38.4Kbps, the size of the 10 channel interleaver 100 is doubled to N=3072 (=2×1536).

When the data rate increases from 20Kbps to 38.4Kbps as stated above, null data is written by an upper layer in an empty interval corresponding to an interval excepting the 20Kbps×20msec period out of the data symbols input to a 15 channel encoder (not shown). That is, the channel interleaver with size N writes $(38.4-20)/38.4=47.92\%$ of its output with null data before transmission. Therefore, from the viewpoint of reception symbol energy E_s , it can be presumed that 47.92% of energy is lost. The reason that an energy loss occurs is because the physical layer has no way to process the null data in the non-FDRT scheme. 20 Even though the null data is subjected to symbol repetition before transmission, a forward supplemental channel (F-SCH) scheme has a disadvantage that it cannot perform symbol combining. Further, since the null data is different according to a data rate of the input data, the upper layer must previously send the null data to the base station and the mobile station. In addition, energy of the null data must 25 be restored before the null data passes through a channel decoder, and the upper layers L1/L2 process only the decoded information symbols after the channel decoder, thus causing a decrease in decoding performance.

The FDRT scheme has been proposed to solve the problem and improve

the performance of the non-FDRT scheme. Active research has been made on the FDRT rate matching technique for increasing the data transmission efficiency of the channel coding scheme and improving the system performance in a multiple access and multi-channel system using the channel coding scheme. The 5 principles of the FDRT technique are based on the assumption that the used channel code is a convolutional code, a linear block code, or a convolutional code using a concatenated code. In particular, for the 3GPP (3rd Generation Project Partnership) IS-2000 air interface, the FDRT rate matching technique has been provisionally determined as the standard specification to increase the data 10 rate efficiency of the channel coding scheme and improve the system performance in the multiple access and multi-channel system, and research is presently being conducted on the implementation of this technique.

FIG. 3 illustrates a structure of a flexible data rate transmission (FDRT) 15 rate matching device according to the prior art.

Prior to describing FIG. 3, various terminologies used herein are defined in Table 1 below. That is, $c[n]$, $d[n]$, $f[n]$ and $r[n]$ in FIG. 3 each indicate the data symbols defined in Table 1. Here, the “symbol” is expressed with one bit having 20 a value of ‘1’ or ‘0’. In general, the symbol is comprised of one or more bits. Herein, however, every data bit expressed with one bit will be referred to as a “symbol”.

Table 1

Terms	Definitions
$c[n]$	Coded symbols (0..L-1) from Channel encoder
$r[n]$	Repeated coded symbols (0..LM-1) by repetition
$f[n]$	Punctured coded symbols (0..N-1) by FDRT
$d[n]$	Interleaved coded symbols (0..N-1) by Channel interleaver

In Table 1, $c[n]$ indicates coded symbols output from the channel encoder (not shown), and $r[n]$ indicates coded symbols repeated by a repeater 110. Further, $f[n]$ indicates coded symbols punctured by a puncturer 120 out of 5 the repeated coded symbols, and $f[n]$ indicates coded symbols interleaved by the interleaver 100 out of the punctured coded symbols. The channel encoder outputs a stream (or sequence) of L coded symbols. The repeater 110 repeats the L coded symbols M times and outputs LM symbols. The puncturer 120 punctures P symbols out of the LM repeated coded symbols, and thus outputs N FDRT-10 processed symbols. The channel interleaver 100 interleaves a stream of the N FDRT-processed symbols.

For reference, since $L \leq N$ in the FDRT scheme, the input coded symbols are always subjected to repetition. This is because the FDRT scheme is so 15 designed as to guarantee a data rate of input transmission data to be matched with an IS-2000 channel interleaver size. Hence, the FDRT scheme includes the puncturer used for matching the interleaver size $N = LM - P$ after repetition, so that the number of transmitted symbols is fundamentally higher than the number L of the coded symbols.

20

Referring to FIG. 3, when the number L of the coded symbols is smaller than the channel interleaver size N , the repeater 110 repeats the coded symbols M times. In case of an IS-2000 system, since the channel interleaver size increases/decreases a multiple of 2 according to a spreading factor (SF), M 25 becomes 2 at least. Since the number of the coded symbols repeated by the repeater 110 is larger than N , the puncturer 120 performs puncturing in order to match the number of the repeated coded symbols to the size N of the channel interleaver 100.

Proprietary information
Hewlett-Packard
Document ID: 100000000000000
Release date: 1999-08-01
Version: 1.0
Page: 5 of 5

FIGs. 4A to 4D illustrate a format of a coded symbol frame reassembled by the repeater 110 and the puncturer 120 in the flexible data rate transmission (FDRT) matching device shown in FIG. 3.

5 Specifically, FIG. 4A illustrates L coded symbols within one frame, and FIG. 4B illustrates LM coded symbols repeated M times by the repeater 110. Further, FIG. 4C illustrates the LM coded symbols, where N coded symbols are to be interleaved by the channel interleaver 100 and LM-N coded symbols are to be punctured by the puncturer 120. Here, the LM-N coded symbols are 10 distributed such that the symbols should be uniformly punctured within the frame at intervals of D. Finally, FIG. 4D illustrates the coded symbols after puncturing, and the resulting coded symbols are provided to the channel interleaver 100 for channel interleaving.

15 Referring to FIGs. 4A to 4D, the reassembled coded symbol frame will be compared with the non-FDRT coded symbol frame shown in FIG. 2. In the FDRT scheme, there is no null data within the frame and every symbol is processed as a coded symbol. By using the FDRT scheme rather than the non-FDRT scheme, the receiver can increase energy of the coded symbol received at 20 the same transmission power. The coded symbol energy refers to the energy of the coded symbols after symbol combining. In this way, it is possible to decrease the transmission power of the base station, required in guaranteeing the same QoS (Quality of Service), thereby causing an increase in the channel capacity.

25 In FIG. 4C, black blocks indicate the symbols to be punctured and 'D' indicates a puncturing distance. The puncturing distance D is a parameter for determining a puncturing method performed to output N symbols from LM symbols. An FDRT algorithm is used to specify the relationships among the parameters L, M, N, P and D.

Table 2 below discloses the FDRT algorithm defined in the IS-2000 specification. In the following description, the FDRT algorithm will be described using the original terminologies excerpted from the original document, for convenience of explanation.

5

Table 2

If variable-rate Reverse Supplemental Channel operation, flexible data rates, or both are supported, puncturing after symbol repetition is calculated as described here. However, the puncturing in 3.1.3.1.6.1 and 3.1.3.1.6.2 is used for the frame formats listed in Table 3.1.3.10.2-1 for the Forward Dedicated Control Channel, Table 3.1.3.11.2-1 for the Forward Fundamental Channel, or Tables 3.1.3.12.2-1, 3.1.3.12.2-2, or 3.1.3.12.2-3 for the Forward Supplemental Channel.

The number of repeated symbols punctured per frame is defined by

$$P = LM - N$$

where L = Number of specified encoded symbols per frame at encoder output

N = Desired channel interleaver size ($N \geq L$)

$M = \lceil N/L \rceil$ is the symbol repetition factor for flexible data rate

If P is equal to 0, then puncturing is not required. If puncturing is necessary, every D^{th} repeated symbol is deleted until the required number of punctured symbols per frame, P , is achieved. That is, if the unpunctured symbols are numbered from 1 to LM , then symbols numbered $D, 2D, 3D, \dots$ are deleted.

$D = \lfloor LM/P \rfloor$ for $P > 0$; otherwise, puncturing is not required.

As shown in the FDRT algorithm of Table 2, the parameter D is determined from given parameters L and N , and then, every D^{th} coded symbol is punctured from the first coded symbol using the determined parameter value D ,

thereby finally puncturing $P=LM-N$ coded symbols. However, since the FDRT does not consider the following conditions in view of the characteristics of the convolutional code, it may have a performance degradation problem.

5 The convolutional code and the linear block code using a single decoder are generally used for the channel coding scheme. In this case, the following conditions should be fully considered and reflected during puncturing in the FDRT scheme for increasing a data transmission efficiency of the channel encoding scheme and improving the system performance in the multiple access
10 and multi-channel system using the channel encoding scheme.

Condition (1): An input symbol sequence is punctured with a puncturing pattern having a specific period;

15 Condition (2): The number of the punctured bits of the input symbols is minimized, if possible; and

Condition (3): The coded symbols output from an encoder are punctured using a uniform puncturing pattern.

20 The foregoing conditions are based on the assumption that error sensitivity of the coded symbols output from the channel encoder is almost similar with respect to every symbol in one frame (or codeword). When data is actually transmitted in the FDRT mode, it is possible to obtain an affirmative
25 outcome by using the above conditions as major puncturing restriction factors. However, in most cases, the IS-2000 FDRT scheme does not satisfy the above conditions.

FIG. 5 illustrates how the FDRT device shown in FIG. 3 punctures the
30 coded symbols before transmission. Specifically, FIG. 5 illustrates a puncturing

pattern used when transmitting 15Kbps symbols at RC3 data rate = 19.2Kbps in the FDRT mode. That is, FIG. 5 is a diagram for explaining a problem that may occur when the foregoing conditions are not satisfied. A condition used in FIG. 5 is shown in Table 3 below.

5

Table 3

IS-2000 RC3 (Code rate R=1/4)
Maximum Assigned Data Rate = 19.2kbps N=1536bits
Input Data Rate = 15kbps
Coded symbols per frame (L) = 1200bits
$M = \lceil N/L \rceil = \lceil 1536/1200 \rceil = 2$
P = 864bits (LM-N=2400-1536)
$D = \lfloor LM/P \rfloor = \lfloor 2400/864 \rfloor = \lfloor 2.778 \rfloor = 2$

Referring to FIG. 5, it is noted that puncturing is actually performed only at the leading 1728 bits of the coded symbol frame and not performed in the 10 following 672-bit interval of the frame. For reference, in FIG. 5, the black blocks indicate the punctured symbols and the dotted blocks indicate 672 symbols which are repeated twice before transmission. The leading 1728 twice-repeated symbols are selectively transmitted every other symbol. In this method, N=1536 (=864+672) symbols are formed (or assembled). The N=1536-bit format in the 15 frame violates the Condition (3) above. Therefore, the FDRT scheme may have a performance degradation problem due to non-uniform puncturing.

FIG. 6 is a diagram for explaining the problem of the conventional FDRT scheme. Specifically, FIG. 6 illustrates distribution of the symbol energy and the 20 number of symbols per unit frame at the final stage of the receiver.

Referring to FIG. 6, a channel receiver 200 receives the symbols

transmitted in the FDRT mode and provides the received symbols to an erasure insertion and symbol combining part 210. FIG. 6 shows the relative distribution of the symbol energy Es for the respective symbols when the symbol combining part 210 performs symbol combining on the provided symbols. As illustrated, 5 when the symbol energy Es of 864 unrepeated symbols is generalized to 1.0, the following 672 repeated symbols are subjected to symbol combining with M=2, making Es become 2.0. Therefore, the tail symbols have an average gain of Es/No=+3dB in the same channel environment. That is, an R=1/4 channel decoder 220 decodes the non-uniformly distributed 1200 symbols and outputs 10 300 information symbols. As will be described later with reference to FIGs. 12 and 13, it is noted from the simulation results that the conventional FDRT device has considerable performance degradation.

The occurrence of the non-uniform puncturing is caused by the value D 15 which determines the puncturing pattern. That is, when LM/P is not an integer, the conventional IS-2000 FDRT algorithm defines the value D as $\lfloor LM/P \rfloor$ indicating a maximum integer smaller than LM/P. In this case, only P×D symbols are actually punctured, and the puncturing is not performed in the remaining P×(LM/P-D)-bit interval. In FIG. 5, for example, since LM/P=2400/864=2.778, 20 D=2 and LM/P-D=0.778. Therefore, the puncturing is performed in the P×D=864×2=1728-bit interval, while the puncturing is not performed in the P×(LM/P-D)=864×0.778=672-bit interval. In conclusion, the non-uniform puncturing occurs due to a difference of (LM/P-D) in the process of determining the value D.

25

The conventional FDRT scheme has the following disadvantages:

- 1) The FDRT scheme using the convolutional code or liner block code requires a uniform puncturing scheme in view of the property that error sensitivity of coded symbols output from the channel encoder is almost similar

with respect to every symbol within one frame (or codeword). However, since such an assumption is not conducted in the existing IS-2000 FDRT scheme, it is necessary to modify the existing FDRT scheme.

2) From the viewpoint of symbol repetition, the existing IS-2000 FDRT scheme fundamentally regards the FDRT scheme as a repetition scheme, considering that the puncturing pattern is not affected greatly. However, this should be interpreted in the same context as the puncturing. That is, a uniform repetition scheme should be used for the FDRT scheme with optimal performance even in case of the repetition, in view of the property that error sensitivity of the coded symbols output from the channel encoder is almost similar with respect to every symbol within one frame (or codeword). However, since such an assumption is not conducted in the existing IS-2000 FDRT scheme, it is necessary to modify the existing FDRT scheme.

15

SUMMARY OF THE INVENTION

It is, therefore, an object of the present invention to provide an apparatus and method for guaranteeing optimal performance without performance degradation when matching a frame having coded symbols flexibly determined according to variation of a data rate to an interleaver size in a data communication system.

It is another object of the present invention to provide a flexible data rate transmission (FDRT) apparatus and method which flexibly operates according to a data rate by simply adjusting its structure and initial setting value in a data communication system using a convolutional code or a linear block code.

To achieve the above and other objects, there is provided a method for generating a stream of N symbols by puncturing a stream of repeated symbols in a system including an encoder for generating a stream of L symbols, a repeater

for repeating the stream of L symbols, and a puncturer for puncturing the stream of repeated symbols and generating a stream of N symbols, where N is larger than L. The method comprises generating a stream of LM repeated symbols by repeating the stream of L symbols M times, where M is an minimum integer
 5 larger than N/L ; calculating a first puncturing interval D1 defined as a minimum integer larger than LM/P for a number, $P=LM-N$, of symbols to be punctured, and a first symbol puncturing number P1 defined as a maximum integer smaller than $LM/D1$; calculating a second symbol puncturing number P2 indicating a difference between the number P of the symbols to be punctured and the first
 10 symbol puncturing number P1, and a second puncturing interval D2 defined as $sD1$ for a selected one integer s out of integers smaller than or equal to a maximum integer smaller than $P1/P2$; and generating a stream of N symbols by puncturing the stream of LM repeated symbols at the first puncturing interval D1 and the second puncturing interval D2.

15

BRIEF DESCRIPTION OF THE DRAWINGS

The above and other objects, features and advantages of the present invention will become more apparent from the following detailed description
 20 when taken in conjunction with the accompanying drawings in which:

FIG. 1 is a diagram illustrating a conventional non-FDRT channel interleaver;

FIG. 2 is a diagram illustrating a coded symbol frame format transmitted according to the non-FDRT mode;

25 FIG. 3 is a diagram illustrating a structure of a conventional flexible data rate matching device;

FIGs. 4A to 4D are diagrams illustrating a coded symbol frame format reassembled by a repeater and a puncturer in the flexible data rate matching device shown in FIG. 3;

30 FIG. 5 is a diagram illustrating an example where the coded symbols are

open during recording if write in buffer is full write in buffer if read head is full

punctured by the FDRT matching device shown in FIG. 3;

FIG. 6 is a diagram for explaining a problem of the conventional FDRT scheme, the diagram illustrating distribution of symbol energy and the number of symbols per unit frame at a final stage of a receiver;

5 FIG. 7 is a diagram illustrating an exemplary method of puncturing the coded symbols according to a puncturing pattern proposed in the present invention;

FIGs. 8A and 8B are diagrams illustrating distribution of symbol energy and the number of symbols per unit frame at a final stage of a receiver associated
10 with a flexible data rate matching device according to an embodiment of the present invention;

FIG. 9 is a flow chart illustrating a procedure for performing flexible data rate matching and transmission operations according to a first embodiment of the present invention;

15 FIG. 10 is a diagram illustrating a structure of a flexible data rate matching device according to the first embodiment of the present invention;

FIG. 11 is a diagram illustrating another structure of a flexible data rate matching device according to the first embodiment of the present invention;

20 FIG. 12 is a flow chart illustrating a procedure for performing flexible data rate matching and transmission operations according to a second embodiment of the present invention;

FIG. 13 is a diagram illustrating a structure of a flexible data rate matching device according to the second embodiment of the present invention;

25 FIG. 14 is a diagram illustrating another structure of a flexible data rate matching device according to the second embodiment of the present invention;

FIGs. 15 and 16 are diagrams illustrating a comparison between the simulation results of the flexible data rate matching and transmission operations proposed in the present invention and the simulation results according to the prior art.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

A preferred embodiment of the present invention will be described herein below with reference to the accompanying drawings. In the following 5 description, well-known functions or constructions are not described in detail since they would obscure the invention in unnecessary detail.

The present invention provides an improved FDRT scheme capable of securing uniform puncturing or repetition, thereby solving a problem of the 10 conventional FDRT scheme. To this end, a uniform puncturing pattern or a uniform repetition pattern is required. Therefore, the present invention provides a method for creating a new puncturing pattern for FDRT and then puncturing the coded symbols according to the created new puncturing pattern.

15 The determination of a proper puncturing distance D is important in performing the uniform puncturing or uniform repetition in the FDRT scheme. In other words, the non-uniform puncturing or repetition is caused by the parameter D for determining the puncturing pattern or the repetition pattern. That is, in the conventional IS-2000 FDRT algorithm, when LM/P is not an integer, $\lfloor LM/P \rfloor$ 20 indicating the maximum integer smaller than LM/P is defined as the parameter D. Therefore, in this case, only $P \times D$ bits are actually punctured and the puncturing is not performed in the remaining $P \times (LM/P - D)$ -bit period. For example, since $LM/P = 2.778$, $D = 2$ and $LM/P - D = 0.778$. Therefore, the puncturing is performed in the $P \times D = 864 \times 2 = 1728$ -bit interval, while the puncturing is not 25 performed in the $P \times (LM/P - D) = 864 \times 0.778 = 672$ -bit interval. In conclusion, the non-uniform puncturing occurs due to a difference of $(LM/P - D)$ in the process of determining the value D. To solve this problem, the following basic conditions are introduced, and then an algorithm based on the conditions will be described.

FDRT Condition (1): $P \times D$ determined from L and N should satisfy $P \times D \geq LM$. That is, D should satisfy $D \geq LM/P$. Here, P and D are integers.

FDRT Condition (2): $(P - \lfloor LM/D \rfloor)$ symbols determined from the parameter value D satisfying FDRT Condition (1) are punctured or repeated as uniformly (or at regular intervals) as possible over LM symbols. Here, the determined symbol position should not overlap with the position determined by the parameter D satisfying FDRT Condition (1).

FDRT Condition (3): The non-uniform repetition or puncturing due to a difference of $(LM/P - D)$ in the process of determining the parameter D should be minimized.

Now, an operation of the FDRT scheme according to an embodiment of the present invention will be described, considering the above FDRT conditions. First, a description will be made of an embodiment to which the FDRT algorithm according to the present invention is applied. Subsequently, a generalized FDRT algorithm will be described.

20 New Flexible Data Rate Transmission Algorithm Type 1

A description will be made of an embodiment to which the FDRT algorithm according to the present invention is applied. The conditions used in this embodiment are shown in Table 4 below, and the algorithm is shown in Table 25 5 below.

Referring to Table 4, the embodiment of the present invention is applied to IS-2000 RC3. The maximum assigned data rate is 19.2Kbps, the interleaver size is $N=1536$ bits, and an input data rate is 15Kbps. Further, the number of

coded symbols per frame is $L=1200$ bits. Therefore, the number of repetitions for the $L=1200$ coded symbols becomes $M=2$. The repetition number M is defined as the minimum integer larger than N/L ($=$ (interleaver size)/(coded symbols per frame)). That is, the repetition number M is defined as $M=\lceil N/L \rceil$. The number P of coded symbols to be punctured is determined by subtracting the interleaver size N from the repeated coded symbols LM . The puncturing distance D is defined as $D=\lceil LM/P \rceil$.

Table 4

IS-2000 RC3 (Code rate R=1/4)

Maximum Assigned Data Rate = 19.2kbps [N=1536bits]

$N=1536$ bits

Input Data Rate = 15kbps

Coded symbols per frame (L) = 1200bits

$$M=\lceil N/L \rceil=\lceil 1536/1200 \rceil=2$$

$P = 864$ bits ($LM-N=2400-1536$)

$$D=\lceil LM/P \rceil=\lceil 2400/864 \rceil=\lceil 2.778 \rceil=3$$

10

Table 5

$$D=\lceil LM/P \rceil=3$$

The repeated symbol is deleted if the following condition is satisfied.

If $(k \bmod 3 = 2 \text{ or } k \bmod 36 = 0)$ then Puncturing

where $k=0,1,2,\dots,2399$

Referring to Table 5, in the algorithm according to an embodiment of the present invention, “ $k \bmod(?) 3$ ” indicates a modulo-3 operation of calculating a remainder determined by dividing k by 3. FDRT Condition (1) is used in the process of calculating D , and FDRT Condition (2) is used in the process having a variable ‘36’.

FIG. 7 illustrates an exemplary method of puncturing the coded symbols according to a puncturing pattern proposed in the present invention. This method is based on the condition of Table 4 and the algorithm of Table 5.

5

Referring to FIG. 7, it is noted that the puncturing is actually uniformly performed over the overall interval of the coded symbol frame. In FIG. 7, the black blocks indicate the punctured symbols. Further, it is noted that the symbols transmitted after twice repetition and a selected one of the symbols transmitted
10 after twice repetition are uniformly distributed. Therefore, the N=1536 frame has a symbol format coincident with FDRT Condition (3). Hence, such an FDRT scheme is free from performance degradation because of the uniform puncturing, and has a near-optimal performance.

15

FIGs. 8A and 8B illustrate distribution of symbol energy and the number of symbols per unit frame at the final stage of a receiver associated with a flexible data rate matching device according to an embodiment of the present invention.

20

Referring to FIGs. 8A and 8B, a channel receiver 200 receives the symbols transmitted in the inventive FDRT mode and provides the received symbols to an erasure insertion and symbol combining part 210. The symbol combining part 210 outputs 1200 symbols, as shown in FIG. 8A, and the output symbols have the relative symbol energy distribution shown in FIG. 8B. As
25 illustrated, when the symbol energy Es of 864 unpeated symbols is generalized to 1.0, the following 672 repeated symbols are subjected to symbol combining with M=2, making Es become 2.0. FIG. 8B shows that the symbols are uniformly distributed over the entire interval. The uniform symbol distribution contributes to performance improvement of a channel decoder 220, for which a Viterbi
30 decoder is typically used.

Generalized Flexible Data Rate Transmission Algorithm GFDRTA-I

A generalized flexible data rate transmission algorithm according to the 5 present invention will be described. The FDRT algorithm and the parameters used in the algorithm are defined in Table 6.

Table 6

$$P = LM - N$$

where L = Number of specified encoded symbols per frame at encoder output

N = Desired channel interleaver size ($N \geq L$)

M = $\lceil N/L \rceil$ is the symbol repetition factor for flexible data rate

- 10 In table 6, L indicates the number of coded symbols per frame, out of streams of the coded symbols output from the encoder. Further, N indicates a predetermined channel interleaver size, and is defined as a value larger than or equal to the number L of the coded symbols per frame. In addition, M indicates the number of repetitions for the coded symbols, and is defined as $\lceil N/L \rceil$. That is, 15 the repetition number M is defined as the minimum integer larger than N/L . Therefore, the number P of the coded symbols to be punctured is defined as P=LM-N.

As a first embodiment, in the algorithm of Table 6, if P=0, then the 20 puncturing is not performed. In the puncturing process, every D1th symbol and every (D2+1)th symbol (where D2 is an even number) out of the LM coded symbols are punctured, until P symbols are punctured per unit frame. That is, when the LM coded symbols are ordered from 1 to LM, the D1th, 2D1th, 3D1th,... coded symbols and the (D2+1)th, (2D2+1)th, (3D2+1)th,... coded symbols (where

D₂ is an even number) are punctured. Here, the (D₂+1)th, (2D₂+1)th, (3D₂+1)th,... coded symbols are punctured so as not to overlap with the mD₁th (where m=1,2,3,...) coded symbols in terms of puncturing positions. Therefore, if necessary, it is possible to consider another method for preventing the (D₂+1)th, 5 (2D₂+1)th, (3D₂+1)th,... coded symbols from overlapping with the mD₁th (where m=1,2,3,...) coded symbols. For example, it is also possible to puncture (D₂-1)th, (2D₂-1)th, (3D₂-1)th,... coded symbols (where D₂ is an odd number) instead of the (D₂+1)th, (2D₂+1)th, (3D₂+1)th,... coded symbols. Even in this case, the (D₂-1)th, (2D₂-1)th, (3D₂-1)th,... coded symbols are punctured so as not to overlap 10 with the mD₁th (where m=1,2,3,...) coded symbols in terms of puncturing positions. That is, D₁ and D₂ indicate puncturing distance values for determining the distances among the P symbols to be punctured out of the LM repeated coded symbols. D₁ and D₂ used herein are defined by Equation (1) below.

15 As a second embodiment, in the algorithm of FIG. 6, if P=0, then the puncturing is not performed. In the puncturing process, every D₁th symbol and every ((a multiple of D₂)-D₂+ $\lfloor D_1/2 \rfloor$)th symbol out of the LM coded symbols are punctured, until P symbols are punctured per unit frame. That is, when the LM coded symbols are ordered from 1 to LM, the D₁th, 2D₁th, 3D₁th,... coded 20 symbols and the (D₂+ $\lfloor D_1/2 \rfloor$)th, (2D₂+ $\lfloor D_1/2 \rfloor$)th, (3D₂+ $\lfloor D_1/2 \rfloor$)th,... coded symbols are punctured. Here, the (D₂+ $\lfloor D_1/2 \rfloor$)th, (2D₂+ $\lfloor D_1/2 \rfloor$)th, (3D₂+ $\lfloor D_1/2 \rfloor$)th,... coded symbols are punctured so as not to overlap with the mD₁th (where m=1,2,3,...) coded symbols in terms of puncturing positions. D₁ and D₂ used herein are also defined by Equation (1) below.

25

Equation (1)

D₁ = $\lceil LM/P \rceil$ for P > 0: Otherwise, puncturing is not required.

P₁ = $\lfloor LM/D_1 \rfloor$

P₂ = P-P₁

$D_2 = sD_1$ for $P_2 > 0$: Otherwise, puncturing is nor required.

In Equation (1), s indicates the maximum integer out of integers within a range satisfying Equation (2) below.

5

Equation (2)

$$(sD_1) \leq \lfloor LM/P_2 \rfloor,$$

$$s \leq (\lfloor LM/P_2 \rfloor)/D_1 = (\lfloor LM/D_1 \rfloor)/P_2 = \lfloor P_1/P_2 \rfloor$$

10 Referring to Equations (1) and (2), the puncturing distance (or interval) D_1 is defined as the minimum integer larger than LM/P for the number, $P=LM-N$, of the remaining symbols to be punctured. P_1 indicates the symbol puncturing number and is defined as the maximum integer smaller than LM/D_1 . P_2 indicates a symbol puncturing number determined by a difference between the total 15 number P of the symbols to be punctured and the symbol puncturing number P_1 . The puncturing distance D_2 is defined as sD_1 for an integer ‘ s ’ out of integers smaller than or equal to the maximum integer smaller than P_1/P_2 .

In Table 6 and Equations (1) and (2), in order to match a stream of L 20 ($<N$) coded symbols to the interleaver size N, the stream of the L coded symbols is repeated M times thus generating a stream of LM coded symbols, and the stream of the LM repeated coded symbols is punctured at the first puncturing interval D_1 and the second puncturing interval D_2 according to a first puncturing pattern A and a second puncturing pattern B. Here, the first puncturing pattern A 25 is defined as a multiple of the first puncturing distance D_1 , while the second puncturing pattern B is defined as a multiple of the second puncturing distance D_2 plus an offset. In the first embodiment, the offset is +1 or -1 ($\text{offset}=\pm 1$). In the second embodiment, the offset is either a value determined by subtracting D_2 from the maximum integer smaller than $D_1/2$ (i.e., $\text{offset} = -D_2 + \lfloor D_1/2 \rfloor$) or a

negative value for a value determined by adding D2 to the maximum integer smaller than D1/2 (i.e., offset = -D2- $\lfloor D1/2 \rfloor$). That is, for the stream of the LM repeated coded symbols, P1 symbols located at the first puncturing interval D1 from the initial symbol are first punctured, and then, P2 symbols located at the 5 second puncturing interval D2 plus an offset (D2 + offset) from the initial symbol are punctured. The first puncturing interval D1 and the second puncturing interval D2 are the values for determining the patterns used for puncturing the symbols uniformly distributed in one frame. Therefore, in the first puncturing process, relatively dense puncturing is performed on the stream of the repeated 10 coded symbols constituting one frame, and in the second puncturing process, relatively loose puncturing is performed on the stream of repeated coded symbols.

In other words, for the stream of the LM repeated coded symbols, P1 15 symbols are punctured, and when the remaining number of the coded symbols after puncturing is larger than the interleaver size N, P2 symbols are punctured for the stream of the (LM-P1) repeated coded symbols. As described above, it is assumed that the embodiment of the present invention performs puncturing on the stream of the repeated coded symbols in two separate steps. This is because 20 even though the number of the coded symbols is smaller than the interleaver size, it is possible to match the number of the coded symbols to the interleaver size by performing puncturing on the repeated coded symbols in two separate steps. Therefore, depending on the circumstances, it is also possible to generate the coded symbols, the number of which is matched to the interleaver size N, in only 25 a single step.

FIG. 9 illustrates a procedure for performing flexible data rate matching and transmission operations, shown in Table 6, according to a first embodiment of the present invention.

Referring to FIG. 9, in step 401, the initial parameters N, L, M and P necessary for FDRT are initialized. The number L of the coded symbols constituting the frame and the interleaver size N are determined according to a given data rate, while the repetition number M and the number P of the symbols to be punctured are determined by the formula in Table 6. In step 402, the first puncturing interval D1 and the first puncturing number P1 are calculated in accordance with the formula given in the algorithm. In step 403, the second puncturing interval D2 and the second puncturing number P2 are calculated in accordance with the formula given in the algorithm. After the parameters are all calculated in steps 402 and 403, steps 404 to 411 are performed while sequentially counting k from 1 to LM. At every counting, if it is determined in steps 405 and 406 that k is a multiple of D1 or D2 (where D2 is an even number), or if it is determined in steps 405 and 408 that k is a multiple of D1 or D2 (where D2 is an odd number), then the corresponding k^{th} coded symbols are punctured in step 407 or 409. In step 405, it is determined whether D2 is an even number or an odd number. If it is determined in step 405 that D2 is an even number, it is determined in step 406 whether k is a multiple of D1 or D2. If it is determined in step 406 that k is a multiple of D1, a k^{th} coded symbol is punctured in step 407; otherwise, if it is determined that k is a multiple of D2, a $(k+1)^{\text{th}}$ coded symbol is punctured in step 407. However, if it is determined in step 406 that k is neither a multiple of D1 nor a multiple of D2, the procedure goes to step 410 to increase the value k by +1. If it is determined in step 405 that D2 is not an even number but an odd number, it is determined in step 408 whether k is a multiple of D1 or D2. If it is determined in step 408 that k is a multiple of D1, a k^{th} coded symbol is punctured in step 409; otherwise, if it is determined that k is a multiple of D2, a $(k-1)^{\text{th}}$ coded symbol is punctured in step 409. However, if it is determined in step 408 that k is neither a multiple of D1 nor a multiple of D2, the procedure goes to step 410 to increase the value k by +1. After step 410, it is determined in step 411 whether $k=LM+1$. If so, the process ends. If not, the steps 405 to 411

are repeated until it is determined in step 411 that $k=LM+1$. In this method, a nearly uniform FDRT puncturing pattern is created and puncturing is performed on the stream of the repeated coded symbols according to the created puncturing pattern.

5

In the operation of steps 401-407, 410 and 411, if it is determined that k is a multiple of D_1 or a multiple of D_2 plus 1 (where D_2 is an even number), then the corresponding k^{th} coded symbol is punctured. In the operation of steps 401-405 and 408-411, if k is a multiple of D_1 or a multiple of D_2 minus 1 (where D_2 is an odd number), then corresponding k^{th} coded symbol is punctured. This is to perform the puncturing at the positions inconsistent with the coded symbols corresponding to a multiple of D_1 . That is, the coded symbols corresponding to a multiple of D_2 plus 1 (where D_2 is an even number) or a multiple of D_2 minus 1 (where D_2 is an odd number) are punctured at the different positions inconsistent 15 with the coded symbols punctured at the positions corresponding to a multiple of D_1 .

FIGs. 10 and 11 illustrate structures of the flexible data rate matching and transmission devices according to the first embodiment of the present invention. Specifically, FIG. 10 illustrates a hardware structure of the FDRT algorithm, and FIG. 11 illustrates a software structure of the FDRT algorithm. That is, the FDRT device according to the first embodiment of the present invention can be realized with either a software module such as a digital signal processor (DSP), a central processing unit (CPU) and a micro-processing unit 25 (MPU), as shown in FIG. 11, or a hardware module such as an application specific integrated circuit (ASIC), as shown in FIG. 10.

Referring to FIG. 10, the flexible data rate matching device according to an embodiment of the present invention includes a channel encoder 10, a repeater 30 110, a puncturer 350, a channel interleaver 100, a symbol index generator 310,

modulo operators 320 and 330, and an OR gate (or logical sum operator) 340.

The channel encoder 10 generates a stream of L coded symbols. The repeater 110 repeats the stream of L coded symbols M times, and outputs LM repeated code symbols. Here, M indicates the number of times the stream of the L coded symbols is repeated, and is defined as the minimum integer larger than 5 N/L . That is, $M=\lceil N/L \rceil$. The puncturer 350 performs a puncturing operation in response to a puncturing enable signal PUNC_EN from the OR gate 340. That is, the puncturing enable signal PUNC_EN is a puncturing pattern for determining 10 the puncturing operation of the puncturer 350. The N-symbol stream output from the puncturer 350 is interleaved by the channel interleaver 100 having the interleaver size N.

The symbol index generator 310 sequentially generates indexes 15 indicating the symbols constituting the stream of LM repeated symbols. The symbol index generator 310 can be realized with a counter. The modulo operator 320 receives the index k generated from the symbol index generator 310 and D1, and generates the puncturing enable signal PUNC_EN of ‘1’, when the k^{th} coded symbol corresponds to a coded symbol at a puncturing position. For example, in 20 the modulo operator 320, “when the k^{th} coded symbol corresponds to a coded symbol at a puncturing position” refers to when the k^{th} coded symbol corresponds to a multiple of D1. The modulo operator 330 receives the index k generated from the symbol index generator 310 and D2, and generates the puncturing enable signal PUNC_EN of ‘1’, when the k^{th} coded symbol 25 corresponds to a coded symbol at a puncturing position. For example, in the modulo operator 330, “when the k^{th} coded symbol corresponds to a coded symbol at a puncturing position” refers to when the k^{th} coded symbol corresponds to a multiple of $(D2+1)$ (where D2 is an even number) or a multiple of $(D2-1)$ (where D2 is an odd number). The OR gate 340 generates the puncturing enable signal

PUNC_EN by ORing the outputs of the modulo operators 320 and 330, and provides the generated puncturing enable signal PUNC_EN to the puncturer 350.

D1 and D2, as described with reference to Table 6, Equations (1) and (2), 5 and FIG. 9, are the puncturing interval values for determining the interval between the symbols to be punctured in the stream of the coded symbols within one frame. The first puncturing interval D1 is defined as the minimum integer larger than LM/P for the number, P=LM-N, of the symbols to be punctured. The second puncturing interval D2 is defined as sD1 for a selected one integer ‘s’ out 10 of integers smaller than or equal to the maximum integer smaller than P1/P2. Here, P1 indicates the first symbol puncturing number and is defined as the maximum integer smaller than LM/D1. P2 indicates the second symbol puncturing number determined by a difference between the total number P of the symbols to be punctured and the first symbol puncturing number P1. That is, 15 D1=⌈LM/P⌉, P1=⌊LM/D1⌋, P2=P-P1, D2=sD1, and s≤⌊P1/P2⌋. The puncturing intervals D1 and D2 and the symbol puncturing numbers P1 and P2 are provided from a puncturing pattern determiner (not shown). The puncturing pattern determiner, the modulo operators 320 and 330, and the OR gate 340 serve as a puncturing pattern generator for generating a puncturing enable signal for 20 determining a puncturing operation of the puncturer 350.

Referring to FIG. 11, as in the flexible data rate matching device shown in FIG. 10, the flexible data rate matching device according to an embodiment of the present invention includes the channel encoder 10, the repeater 110, the 25 puncturer 350, the channel interleaver 100, and the symbol index generator 310. The flexible data rate matching device shown in FIG. 11 is featured by including a puncturing pattern generator 360 in place of the modulo operators 320 and 330 and the OR gate 340 of FIG. 10. By doing so, the flexible data rate matching device is realized by software. The puncturing pattern generator 360 stores an

address generator module program, and generates the puncturing enable signal '1' when k satisfies a specific condition according to the program. The puncturing pattern generator 360 determines the k^{th} coded symbols corresponding to the case where k is a multiple of D_1 or a multiple of D_2 plus 1 (where D_2 is an even number), to puncture the determined coded symbols. The puncturing pattern generator 360 may also determine the k^{th} coded symbols corresponding to the case where k is a multiple of D_1 , a multiple of D_2 plus 1 (where D_2 is an even number), or a multiple of D_2 minus 1 (where D_2 is an odd number), to puncture the determined coded symbols. Then, the flexible data rate matching device 10 actually outputs N symbols out of LM symbols, as in the flexible data rate matching device of FIG. 10.

FIG. 12 illustrates a procedure for performing flexible data rate matching and transmission operations, shown in Table 6, according to a second embodiment of the present invention.

Referring to FIG. 12, in step 601, the initial parameters N , L , M and P necessary for FDRT are initialized. The number L of the coded symbols constituting the frame and the interleaver size N are determined according to a given data rate, while the repetition number M and the number P of the symbols to be punctured are determined by the formula in Table 6. In step 602, the first puncturing interval D_1 and the first puncturing number P_1 are calculated in accordance with the formula given in the algorithm. In step 603, the second puncturing interval D_2 and the second puncturing number P_2 are calculated in accordance with the formula given in the algorithm. After the parameters are all calculated in steps 602 and 603, steps 604 to 608 are performed while sequentially counting k from 1 to LM . At every counting, if it is determined in step 605 that k is (a multiple of D_1) or ((a multiple of D_2)- $D_2+\lfloor D_1/2 \rfloor$), then the corresponding k^{th} coded symbols are punctured in step 606. If it is determined in

step 605 that k is neither (a multiple of D_1) nor ((a multiple of D_2)- $D_2+\lfloor D_1/2 \rfloor$), the procedure goes to step 607 to increase the value k by +1. After step 607, it is determined in step 608 whether $k=LM+1$. If so, the process ends. If not, the steps 605 to 608 are repeated until it is determined in step 608 that $k=LM+1$. In 5 this method, an almost uniform FDRT puncturing pattern is created.

In FIG. 12, if it is determined that k is (a multiple of D_1) or ((a multiple of D_2)- $D_2+\lfloor D_1/2 \rfloor$), then the corresponding k^{th} coded symbol is punctured. Alternatively, if k is (a multiple of D_1) or ((a multiple of D_2)- $D_2-\lfloor D_1/2 \rfloor$), then 10 the corresponding k^{th} coded symbol is punctured. This is to perform the puncturing at the positions inconsistent with the coded symbols corresponding to a multiple of D_1 , and also, to prevent the puncturing range from surpassing the range of LM . In addition, this is to keep the puncturing position of D_1 away from the puncturing position of D_2 as wide as possible, as the D_1 value increases more 15 and more. That is, the coded symbols corresponding to ((a multiple of D_2)- $D_2+\lfloor D_1/2 \rfloor$) are punctured at the different positions inconsistent with the coded symbols punctured at the positions corresponding to a multiple of D_1 .

FIGs. 13 and 14 illustrate structures of the flexible data rate matching 20 and transmission devices according to the second embodiment of the present invention. Specifically, FIG. 13 illustrates a hardware structure of the FDRT algorithm, and FIG. 14 illustrates a software structure of the FDRT algorithm. That is, the FDRT device according to the second embodiment of the present invention can be realized with either a software module such as a DSP and a 25 CPU, as shown in FIG. 14, or a hardware module such as an ASIC, as shown in FIG. 13.

Referring to FIG. 13, the flexible data rate matching device according to an embodiment of the present invention includes a channel encoder 10, a repeater

110, a puncturer 550, a channel interleaver 100, a symbol index generator 510, modulo operators 520 and 530, and an OR gate (or logical sum operator) 540.

The channel encoder 10 generates a stream of L coded symbols. The
 5 repeater 110 repeats the stream of L coded symbols M times, and outputs LM repeated code symbols. Here, M indicates the number of times the stream of the L coded symbols is repeated, and is defined as the minimum integer larger than N/L . That is, $M=\lceil N/L \rceil$. The puncturer 550 performs puncturing on the stream of the LM repeated symbol and outputs a stream of N symbols. Specifically, the
 10 puncturer 550 perform a puncturing operation in response to a puncturing enable signal PUNC_EN from the OR gate 540. That is, the puncturing enable signal PUNC_EN is a puncturing pattern for determining the puncturing operation of the puncturer 550. The N-symbol stream output from the puncturer 550 is interleaved by the channel interleaver 100 having the interleaver size N.

15

The symbol index generator 510 sequentially generates indexes indicating the symbols constituting the stream of LM repeated symbols. The symbol index generator 510 can be realized with a counter. The modulo operator 520 receives the index k generated from the symbol index generator 510 and D1,
 20 and generates the puncturing enable signal PUNC_EN of ‘1’, when the k^{th} coded symbol corresponds to a coded symbol at a puncturing position. For example, in the modulo operator 520, “when the k^{th} coded symbol corresponds to a coded symbol at a puncturing position” refers to when the k^{th} coded symbol corresponds to a multiple of D1. The modulo operator 530 receives the index k
 25 generated from the symbol index generator 510 and D2, and generates the puncturing enable signal PUNC_EN of ‘1’, when the k^{th} coded symbol corresponds to a coded symbol at a puncturing position. For example, in the modulo operator 530, “when the k^{th} coded symbol corresponds to a coded symbol at a puncturing position” refers to when the k^{th} coded symbol corresponds to ((a

multiple of D2)-D2+ $\lfloor D1/2 \rfloor$). The OR gate 540 generates the puncturing enable signal PUNC_EN by ORing the outputs of the modulo operators 520 and 530, and provides the generated puncturing enable signal PUNC_EN to the puncturer 550.

5

D1 and D2, as described with reference to Table 6, Equations (1) and (2), and FIG. 9, are the puncturing interval values for determining the interval between the symbols to be punctured in the stream of the coded symbols within one frame. The first puncturing interval D1 is defined as the minimum integer 10 larger than LM/P for the number, P=LM-N, of the symbols to be punctured. The second puncturing interval D2 is defined as sD1 for a selected one integer ‘s’ out of integers smaller than or equal to the maximum integer smaller than P1/P2. Here, P1 indicates the first symbol puncturing number and is defined as the maximum integer smaller than LM/D1. P2 indicates the second symbol 15 puncturing number determined by a difference between the total number P of the symbols to be punctured and the first symbol puncturing number P1. That is, D1= $\lceil LM/P \rceil$, P1= $\lfloor LM/D1 \rfloor$, P2=P-P1, D2=sD1, and $s \leq \lfloor P1/P2 \rfloor$. The puncturing intervals D1 and D2 and the symbol puncturing numbers P1 and P2 are provided from a puncturing pattern determiner (not shown). The puncturing pattern 20 determiner, the modulo operators 520 and 530, and the OR gate 540 serve as a puncturing pattern generator for generating a puncturing enable signal for determining a puncturing operation of the puncturer 550.

Referring to FIG. 14, as in the flexible data rate matching device shown 25 in FIG. 13, the flexible data rate matching device according to an embodiment of the present invention includes the channel encoder 10, the repeater 110, the puncturer 550, the channel interleaver 100, and the symbol index generator 510. The flexible data rate matching device shown in FIG. 14 is featured by including a puncturing pattern generator 560 in place of the modulo operators 520 and 530

and the OR gates 540 of FIG. 10. By doing so, the flexible data rate matching device is realized by software. The puncturing pattern generator 560 stores an address generator module program, and generates the puncturing enable signal PUNC_EN of '1' when k satisfies a specific condition according to the program.

- 5 The puncturing pattern generator 560 determines the k^{th} coded symbols corresponding to the case where k is (a multiple of D1) or ((a multiple of D2)-D2+ $\lfloor D1/2 \rfloor$), to puncture the determined coded symbols. The puncturing pattern generator 560 may also determine the k^{th} coded symbols corresponding to the case where k is (a multiple of D1) or ((a multiple of D2)-D2- $\lfloor D1/2 \rfloor$), to puncture
- 10 the determined coded symbols. Then, the flexible data rate matching device actually outputs N symbols out of LM symbols, as in the flexible data rate matching device of FIG. 13.

Performance Analysis

15

In the following description, a change in performance according to puncturing of the coded symbols encoded with the convolutional code is analyzed, and an average performance change of a convolutional code with a code rate R according to a puncturing rate and a repetition rate is described. From 20 this, it is possible to predict a performance difference between the conventional IS-2000 FDRT algorithm and the novel FDRT algorithm proposed in the invention, and an average performance value.

First, the reference letters used herein will be defined as follows:

25

R: a code rate of the convolutional code, $R=k/n$;

Rst: (data rate of coded symbols actually transmitted over the channel) \times R, $Rst=NR(\text{bits/sec})$; and

Rfdrt: (data rate of coded symbol output from the channel encoder in the FDRT mode) \times R, $Rfdrt=LR(\text{bits/sec})$.

When the uniform puncturing or repetition pattern is used, a performance change caused by the puncturing or repetition is given by Equation (3) below. Here, when $R_{fdrt} < R_{st}$, the FDRT scheme performs symbol repetition, so that the 5 performance, i.e., coding gain, is improved. However, in contrast, when $R_{fdrt} > R_{st}$, the FDRT scheme performs symbol puncturing, so that the performance, i.e., coding gain, is degraded. As mentioned before, however, since $N > L$, the FDRT scheme generally performs symbol repetition, thus increasing the performance, i.e., coding gain. The point at issue is how the coding gain can 10 be increased according to the pattern.

Equation (3)

$$\text{Average Coding Gain} = 10\log_{10} (R_{st}/R_{fdrt}) \text{ dB}$$

15 For example, when $R_{st}=19.2\text{Kbps}$, the coding gains according to R_{fdrt} are shown in Table 7. Therefore, if the puncturing pattern or the repetition pattern is properly determined and the FDRT algorithm is used, the coding gains shown in Table 7 should be guaranteed.

20 Table 7

	R_{st}	R_{fdrt}	Average Coding Gain
CASE 1	19.2Kbps	17.5Kbps	0.40 (dB)
CASE 2	19.2Kbps	15Kbps	1.07 (dB)
CASE 3	19.2Kbps	10Kbps	2.83 (dB)

FIGs. 15 and 16 illustrate a comparison between the simulation results of the novel FDRT algorithm and the simulation results of the conventional FDRT algorithm.

FIG. 15 is a graph illustrating the simulation results for the case where the present invention is applied to the IS-2000 RC3 (Code Rate R=1/4). This graph is obtained under the following simulation environment. Case (1), Case (2) and Case (3) are given the simulation environments shown in Tables 8, 9 and 10, respectively. In Case (1), a data rate is 15Kbps, the number of coded symbols per frame is L=1200, and the interleaver size is N=1536. Here, 15k_BER_IS2000 and 15k_FER_IS200 indicate the simulation results according to the prior art, while 15k_BER_SEC and 15k_FER_SEC indicate the simulation results according to the present invention. In Case (2), a data rate is 10Kbps, the number of coded symbols per frame is L=800, and the interleaver size is N=1536. In this case, only the simulation results according to the prior art are shown. In Case (3), a data rate is 19.2Kbps. In this case, there occurs no symbol puncturing/repetition.

15 Table 8

CASE (1)

- Data Rate 15kbps (Pure info + CRC + Tail bit = 300 bit)
 - L (Encoded size) = 1200
 - M = 2, N (Channel Interleaver Size) = 1536,
 - P (Num of Puncturing) = $L \times M - N = 864$
 - D (Puncturing Depth) = $\lfloor (L \times M) / P \rfloor = 2$
 - Puncturing Pattern
 - Comparison Subject: 15 kbps signal (Same Puncturing Symbol Number)
 - Puncturing Pattern
- 15KBER_SEC, 15KFER_SEC: NEW Algorithm Type 1 Used
- if ($k \% 3 = 2 \parallel k \% 36 = 0$) ($k = 0, 1, 2, \dots, 2399$) Then Puncturing
- 15KBER_IS2000, 15KFER_IS2000: Old Puncturing Pattern Used

Table 9

CASE (2)

- Data Rate 10kbps (Pure info + CRC + Tail bit = 200 bit)
- L (Encoded size) = 800
- M = 2, N (Channel Interleaver Size) = 1536,
- P (Num of Puncturing) = $L \times M - N = 64$
- D (Puncturing Depth) = $\lfloor (L \times M) / P \rfloor = 25$
- Old Puncturing Pattern Used

Table 10

REFERENCE)

- Reference Curve: 19.2kbps, No Puncturing. No Repetition.

Referring to FIG. 15, as shown in the RC3 simulation results, the FDRT scheme (15k_BER_SEC, and 15k_FER_SEC) according to the present invention provides a gain Eb/No of about 0.9dB to 1.0dB, as compared with the conventional IS-2000 FDRT scheme (15k_BER_IS2000, and 15k_FER_IS2000). This almost approaches the average coding gain 1.07dB, compared with 19.2Kbps, as set forth in Table 7. Such results are obtained by generating the uniform puncturing and repetition pattern, and the performance also shows an almost optimal performance. Therefore, FDRT Condition (1) and FDRT Condition (2) of the FDRT algorithm proposed in the invention play an important role in the performance, and the new FDRT Algorithm Type 1 reflecting the conditions can also provide high performance. However, it is noted that the simulation results for the conventional IS-2000 FDRT algorithm unexpectedly provide a coding gain of about 0.1dB. Such a problem is caused by the asymmetric pattern concentrated at the end of the frame, as described before. In conclusion, in the same channel condition, there occurs a performance difference of about 0.9 to 1.0dB according to the FDRT pattern.

FIG. 16 is a graph illustrating the simulation results for the case where the invention is applied to the RC4 SCH (Code Rate R=1/2). This graph is obtained under the following simulation environment. Case (1), Case (2) and Case (3) are given the simulation environments shown in Tables 11, 12 and 13, 5 respectively. In Case (1), a data rate is 15Kbps, the number of coded symbols per frame is L=600, and the interleaver size is N=768. Here, 15k_BER_IS2000 and 15k_FER_IS200 indicate the simulation results according to the prior art, while 15k_BER_SEC and 15k_FER_SEC indicate the simulation results according to the present invention. In Case (2), a data rate is 17.5Kbps, the number of coded 10 symbols per frame is L=700, and the interleaver size is N=768. In this case, only the simulation results according to the prior art are shown. In Case (3), a data rate is 10Kbps, the number of coded symbols per frame is L=400, and the interleaver size is N=768. In this case, only the simulation results according to the prior art are shown. In Case (4), the data rate is 19.2Kbps. In this case, there occurs no 15 symbol puncturing/repetition.

Table 11

CASE (1)

- Data Rate 15kbps (Pure info + CRC + Tail bit = 300 bit)
- L (Encoded size) = 600
- M = 2, N (Channel Interleaver Size) = 768,
- P (Num of Puncturing) = $L \times M - N = 432$
- D (Puncturing Depth) = $\lfloor (L \times M) / P \rfloor = 2$
- Puncturing Pattern
 - Comparison Subject: 15 kbps signal (Same Puncturing Symbol Number)
 - Puncturing Pattern
- 15KBER_SEC, 15KFER_SEC: NEW Algorithm Type 1 Used
 - if ($k \% 3 = 2 \parallel k \% 36 = 0$) ($k = 0, 1, 2, \dots, 2399$) Then Puncturing
- 15KBER_IS2000, 15KFER_IS2000: Old Puncturing Pattern Used

Table 12

CASE (2)

- Data Rate 17.5kbps (Pure info + CRC + Tail bit = 350 bit)
- L (Encoded size) = 700
- M = 2, N (Channel Interleaver Size) = 768,
- P (Num of Puncturing) = $L \times M - N = 632$
- D (Puncturing Depth) = $\lfloor (L \times M) / P \rfloor = 2$
- Old Puncturing Pattern Used

Table 13

CASE (3)

- Data Rate 10kbps (Pure info + CRC + Tail bit = 200 bit)
- L (Encoded size) = 400
- M = 2, N (Channel Interleaver Size) = 768,
- P (Num of Puncturing) = $L \times M - N = 32$
- D (Puncturing Depth) = $\lfloor (L \times M) / P \rfloor = 25$
- Old Puncturing Pattern Used

5

Referring to FIG. 16, the RC4 simulation results are also equal to the simulation results shown in FIG. 15. As illustrated in FIG. 16, the FDRT scheme (15k_BER_SEC, and 15k_FER_SEC) according to the present invention provides a gain Eb/No of about 0.8dB to 0.9dB, as compared with the conventional IS-2000 FDRT scheme (15k_BER_IS2000, and 15k_FER_IS2000).

What is important next is the 10Kbps performance. In this case, the conventional FDRT algorithm almost approaches the average coding gain 2.83dB shown in Table 7. Such results are obtained because the puncturing distance D for the case of 10Kbps is set to an integer, so that the non-uniform puncturing

due to the difference of LM/P-D does not occur in the process of determining the puncturing distance D. Therefore, this is a good example showing that the performance is linked directly with the prior condition that the difference of LM/P-D should be fully considered in the process of determining the puncturing 5 distance D. Table 14 below shows the simulation environment for this performance.

Table 14

- Data Rate 10kbps (Pure info + CRC + Tail bit = 200 bit)
- L (Encoded size) = 800
- M = 2, N (Channel Interleaver Size) = 1536,
- P (Num of Puncturing) = LM-N = 64
- D (Puncturing Depth) = $\lfloor LM/P \rfloor = 1600/64 = 25 = 25$
- Old Puncturing Pattern Used

10 As described above, the novel FDRT scheme according to the present invention matches a frame having coded symbols flexibly determined according to variation of a data rate to the interleaver size in the data communication system. The FDRT scheme uniformly distributes the puncturing pattern or repetition pattern within the frame by adjusting initial setting values, thereby 15 making it possible to flexibly transmit data according to a data rate without performance degradation.

While the invention has been shown and described with reference to a certain preferred embodiment thereof, it will be understood by those skilled in 20 the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as defined by the appended claims.